

SMARKY NEWS

No 9

31 janvier 1980

PASCAL

Le PASCAL est un langage de programmation récent (développé par N. Wirth à l'EPFZ il y a une dizaine d'années) et qui est de plus en plus utilisé par les micro-ordinateurs.

Ses principaux avantages sont la structuration des données et du programme.

STRUCTURATION DES DONNES

En plus des types de données classiques tels que INTEGER (nombres entiers), REAL (nombres en virgule flottante), BOOLEAN (variables logiques valant TRUE ou FALSE), l'utilisateur peut définir de nouveaux types suivant ses besoins.

Si par exemple on veut utiliser un type 'couleur' pouvant correspondre à 'rouge', 'bleu' ou 'vert', il suffit de faire:

```
TYPE COULEUR = ( ROUGE, BLEU, VERT );
```

au lieu d'assigner un nombre à chaque couleur (rouge = 1, bleu = 2, vert = 3).

Pour assigner 'bleu' à 'couleur', il suffit de faire:

```
COULEUR := BLEU ;
```

ce qui est bien plus clair que "couleur := 2".

Autre avantage: l'instruction "couleur := 15" n'est plus possible!

REFERENCES:

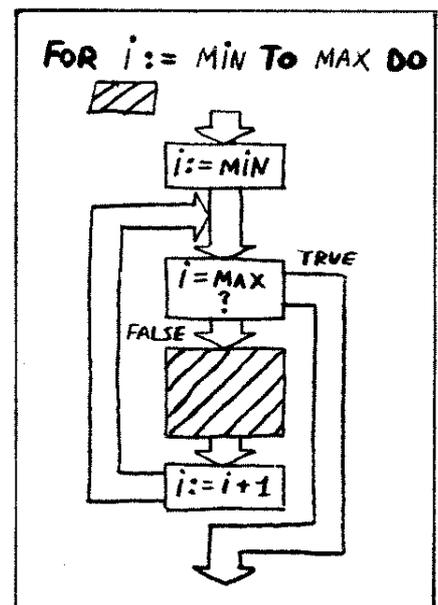
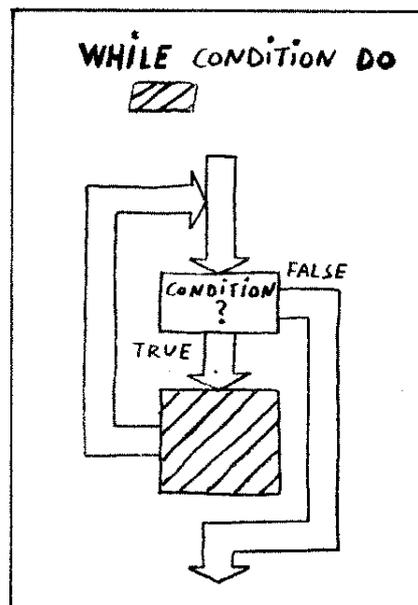
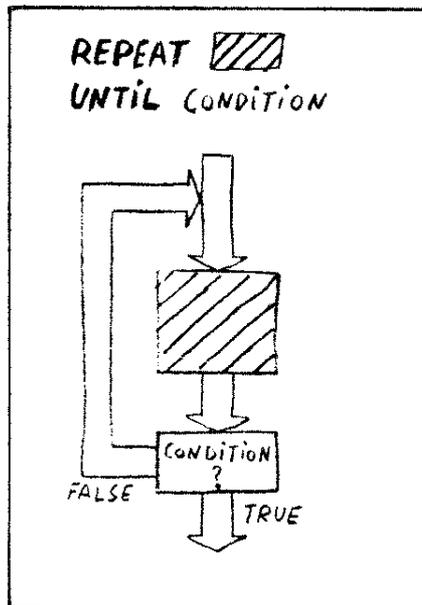
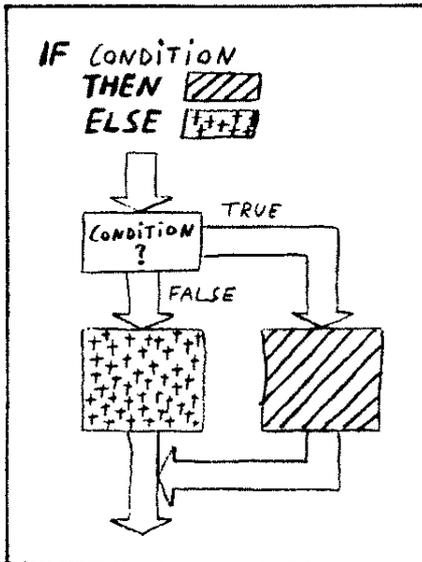
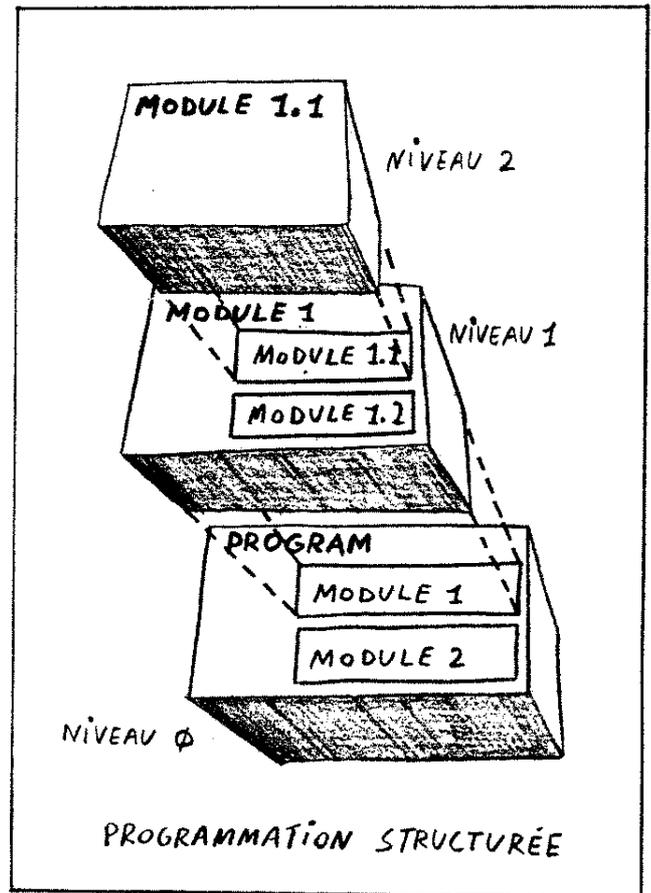
K.Jensen and N. Wirth, PASCAL User Manual and Report, Springer, Berlin, 2nd edition, 1978.

Wakerly, J.F. Microcomputers, Vol.I: architecture and Programming. Wiley, New York, 1979.

STRUCTURATION DU PROGRAMME

Un programme peut être divisé en différents modules ayant chacun une fonction bien définie.

De plus, la conception structurée des algorithmes peut se traduire directement en PASCAL en utilisant les structures de blocs.



Un programme PASCAL comprend dans l'ordre:

- . l'en-tête (donne le nom du programme et définit les fichiers par lesquels il échange des informations avec l'extérieur)
- . les déclarations (des constantes, des variables et des procédures)
- . les instructions proprement dites (simples ou composées) qui décrivent les tâches que le programme accomplit.

Le programme se termine par un point (en général par END.)

EXEMPLES DE PROGRAMMES

- 1) Ecrit sur une ligne les 26 lettres de l'alphabet, un espace, les 26 lettres de l'alphabet dans l'ordre inverse.

```
PROGRAM ALPHABET ( input, output );
CONST
    premier = 'A';
    dernier = 'Z';
VAR
    lettre : char;
BEGIN (* Programme *)
FOR lettre := premier TO dernier DO
    write(lettre);
write(' ');
FOR lettre := dernier DOWNTO premier DO
    write(lettre);
writeln;
END (* Programme *) .
```

- 2) Cherche tous les diviseurs possibles d'un nombre (à l'exception de l'unité).

```
PROGRAM DIVISORS ( input, output );
VAR
    number, divisor : integer;
BEGIN (* Programme *)
REPEAT
    write('Type a number : ');
    readln(number);
    IF number > 0 THEN
        BEGIN
            writeln('The divisors of ', number, ' are :');
            writeln(1);
            FOR divisor := 2 TO number DO
                IF number MOD divisor = 0 THEN
                    writeln(divisor);
            END;
        END;
    UNTIL number <= 0
END (* Programme *) .
```

- 3) Ecrit les puissances de 2 ($2^0=1$ à $2^{12}=4096$) avec leurs équivalents en chiffres romains.

Pour simplifier, le programme écrit IIII au lieu de IV.

```
PROGRAM ROMAN ( input, output );
VAR
    x, y : integer;
BEGIN (* Programme *)
y := 1;
REPEAT
    x := y ; write(x:6, ' ');
    WHILE x >= 1000 DO
        BEGIN write('M') ; x := x - 1000 END;
    IF x >= 500 THEN
        BEGIN write('D') ; x := x - 500 END;
    WHILE x >= 100 DO
        BEGIN write('C') ; x := x - 100 END;
    IF x >= 50 THEN
        BEGIN write('L') ; x := x - 50 END;
    WHILE x >= 10 DO
        BEGIN write('X') ; x := x - 10 END;
    IF x >= 5 THEN
        BEGIN write('V') ; x := x - 5 END;
    WHILE x >= 1 DO
        BEGIN write('I') ; x := x - 1 END;
    writeln ; y := 2*y;
UNTIL y > 5000
END (* Programme *) .
```

IMPLANTATION DU PASCAL UCSD SUR SMAKY6

Le PASCAL de l'UCSD a été implémenté sur SMAKY6 grâce à la fabrication d'une extension mémoire de 32k, amenant ainsi la quantité de RAM adressable à 64k.

Pour un fonctionnement correct, le PASCAL a besoin de:

- . un BIOS (Basic Input Output System) dont la tâche est de gérer le hardware du SMAKY (clavier, écran, floppy, interruptions,...).
- . 1 floppy (2 c'est mieux !) qui contient tous les programmes système (SYSTEM, ...) et les fichiers de travail de l'utilisateur (workfiles).

La mémoire accessible à l'utilisateur commence en 42400 (à la fin de l'écran alphanumérique) et se termine en 177 777. Dans cette zone croissent deux piles, la pile et le stack. La pile contient la zone des variables dynamiques (accessibles avec new, mark, release). Le stack est la pile arithmétique de l'interpréteur.

Les particularités les plus frappantes du PASCAL UCSD sont:

- . la segmentation: découpage logique et physique d'un programme. Seule une partie réside en mémoire, le reste étant sur disque.
- . les bibliothèques: possibilité de mélanger des routines en PASCAL et en assembleur par exemple.

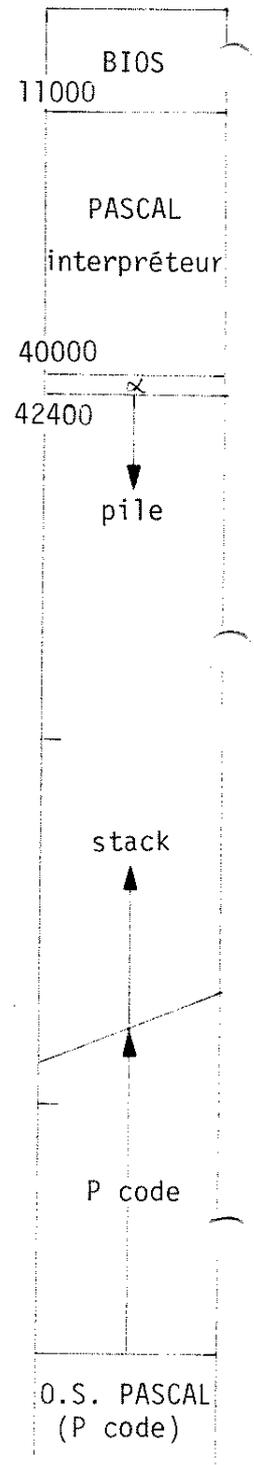
Les périphériques reconnus par le PASCAL sont:

- . CONSOLE: \$DIS et \$KEY (eof = CTRL Z)
- . REMIN: \$PR
- . REMOUT: \$PP (eof = KILL)
- . PRINTER: \$LP
- . # 4: DXØ:
- . # 5: DX1:

Il n'est pour l'instant (sauf si l'on dispose de deux SMAKY) pas possible de transférer un fichier SAMOS sur une disquette PASCAL.

Pour plus d'information, nous vous conseillons:

- . Le PASCAL UCSD (traduction-adaptation du manuel de l'UCSD) par Alain Wegmann
- . W. Findlay, D.A. Watt, PASCAL: An Introduction to Methodical Programming. Pitman, 1978.



ADRESSEZ VOS COMMUNICATIONS A :

EPSITEC-system sa

Chemin de la Mouette, CH - 1092 Belmont